

Getting Started with Python

Baby Steps!

By

Khurram Raheel

What is a Computer Program?

► Suppose you have a problem?

We need a way to find if our friends are studying better than us or not?



Computer Program

Idea

What if we compare our exam results with them!



Computer Program

Lets compare our exam marks
with them

our_marks > their_marks



Computer Program

Lets compare our exam marks
with them

our_marks > their_marks
(800) > (600)



Computer Program

If **our_marks** is greater than **their_marks**
it means we our studying better!



Computer Program

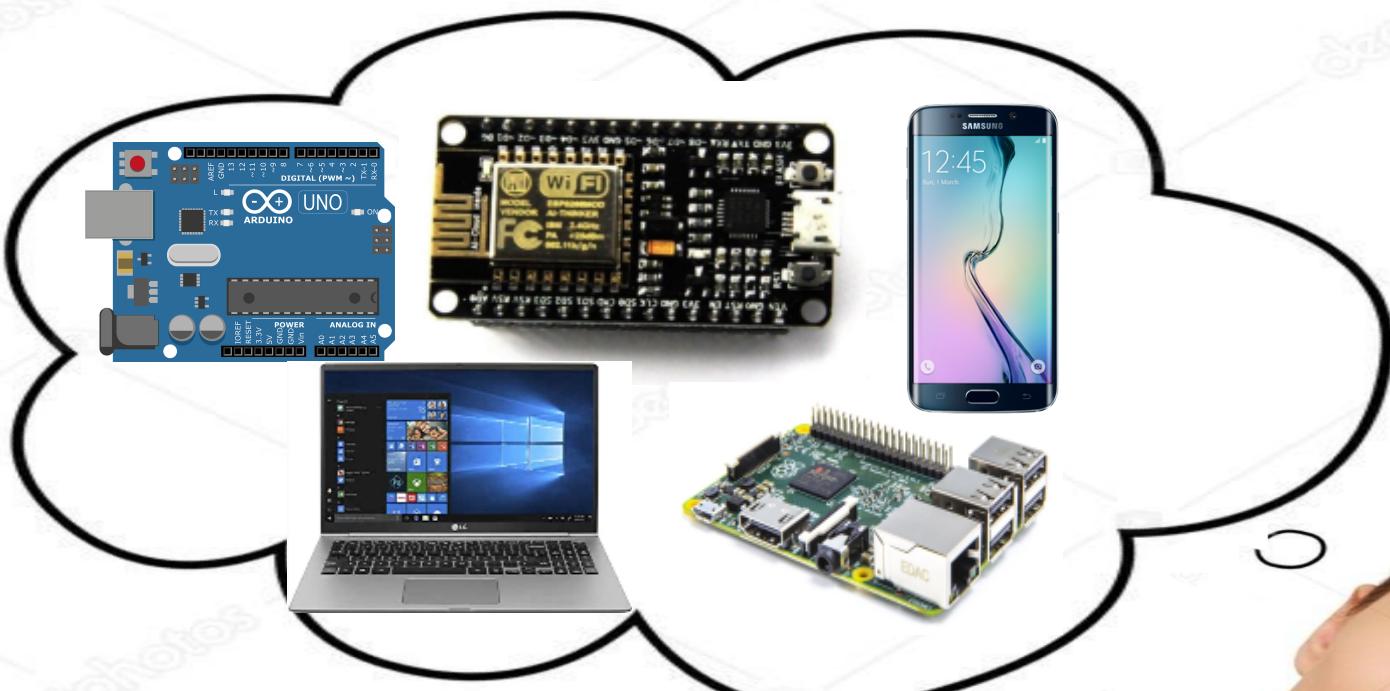
As we have configured the method of finding who is studying better, we can also teach this logic to our devices



Computer Program

- ▶ We can just teach our **programmable device** to understand the input and display the output as per our defined logic.
- ▶ So a computer program is just a sequence of instructions that dictate the flow of predefined commands given by us.

Programmable Device



Getting Started!

► So now we know that what a computer program is, Lets do it now!

Python Welcomes You!



Why Python?

- ▶ Readable and Maintainable Code
- ▶ Multiple Programming Paradigms
- ▶ Compatible with Major Platforms and Systems
- ▶ Robust Standard Library
- ▶ Many Open Source Frameworks and Tools
- ▶ Simplify Complex Software Development
- ▶ Adopt Test Driven Development
- ▶ 1st choice for Data Science driven projects



Multiple Programming Paradigms

- ▶ Website Software Development
- ▶ Desktop apps Development
- ▶ Scientific Computation

Many Open Source Frameworks and Tools

- ▶ Django
- ▶ Flask
- ▶ WebtoPy
- ▶ OpenCV-Python

Tools required for running Python

► Anaconda

Includes machine learning, data mining toolkits and comprehensive playground for developers.



Run Python

► OPEN CMD, TYPE PYTHON

```
C:\WINDOWS\system32\cmd.exe - python
Microsoft Windows [Version 10.0.17134.590]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Pc>python
Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20 2018, 13:35:33) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Understanding Virtual Environments

- ▶ Virtual Environments allow us to have different versions of different libraries in our system without affecting the system.



Magazine Project



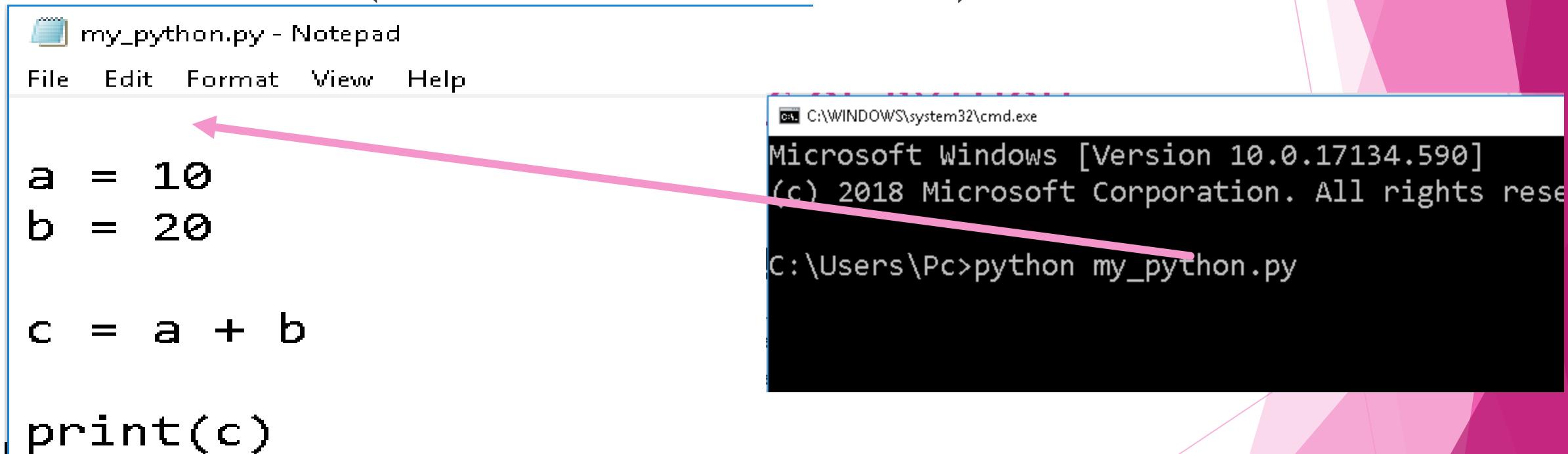
Chat Project



Accounting Software Project

ESSENTIALS OF PYTHON

- We can just start writing python code in a file and run it! (No main function needed).



The image shows a screenshot of a Windows desktop environment. On the left, there is a Notepad window titled "my_python.py - Notepad" containing the following Python code:

```
a = 10
b = 20
c = a + b
print(c)
```

An arrow points from the first line of code ("a = 10") to the cmd.exe window on the right. On the right, there is a Command Prompt window titled "C:\WINDOWS\system32\cmd.exe" with the following text displayed:

```
Microsoft Windows [Version 10.0.17134.590]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Pc>python my_python.py
```

ESSENTIALS OF PYTHON

- ▶ **Variables** (To store Data)
- ▶ **Functions** (To logically group our logics)
- ▶ **Classes** (To model and provide real-world like identification to our data)
- ▶ **Modules** (A simple file for keeping our python definitions and statements apart)

Variables

- ▶ A computer program is always have data!
- ▶ Variable is the most general construct to store data in a computer's memory.



Facebook uses these two variables to store our **username** and **password**

facebook

Email or Phone

Password

Forgotten account?

Log In

Defining a variable in Python

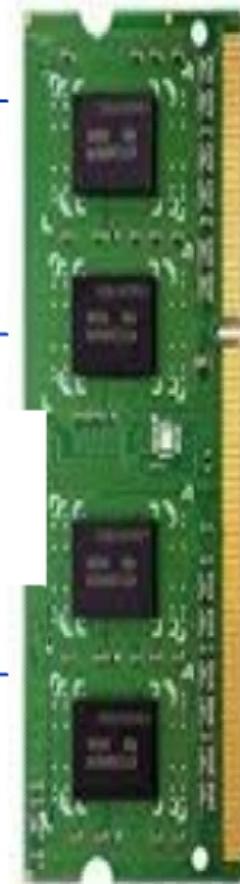
How to Store data? (References;Variable, constants)

```
abc = 10;
```

```
name = "GAMICA";
```

```
c = abc + 20
```

c is now equal to 30 because the abc variable contains 10



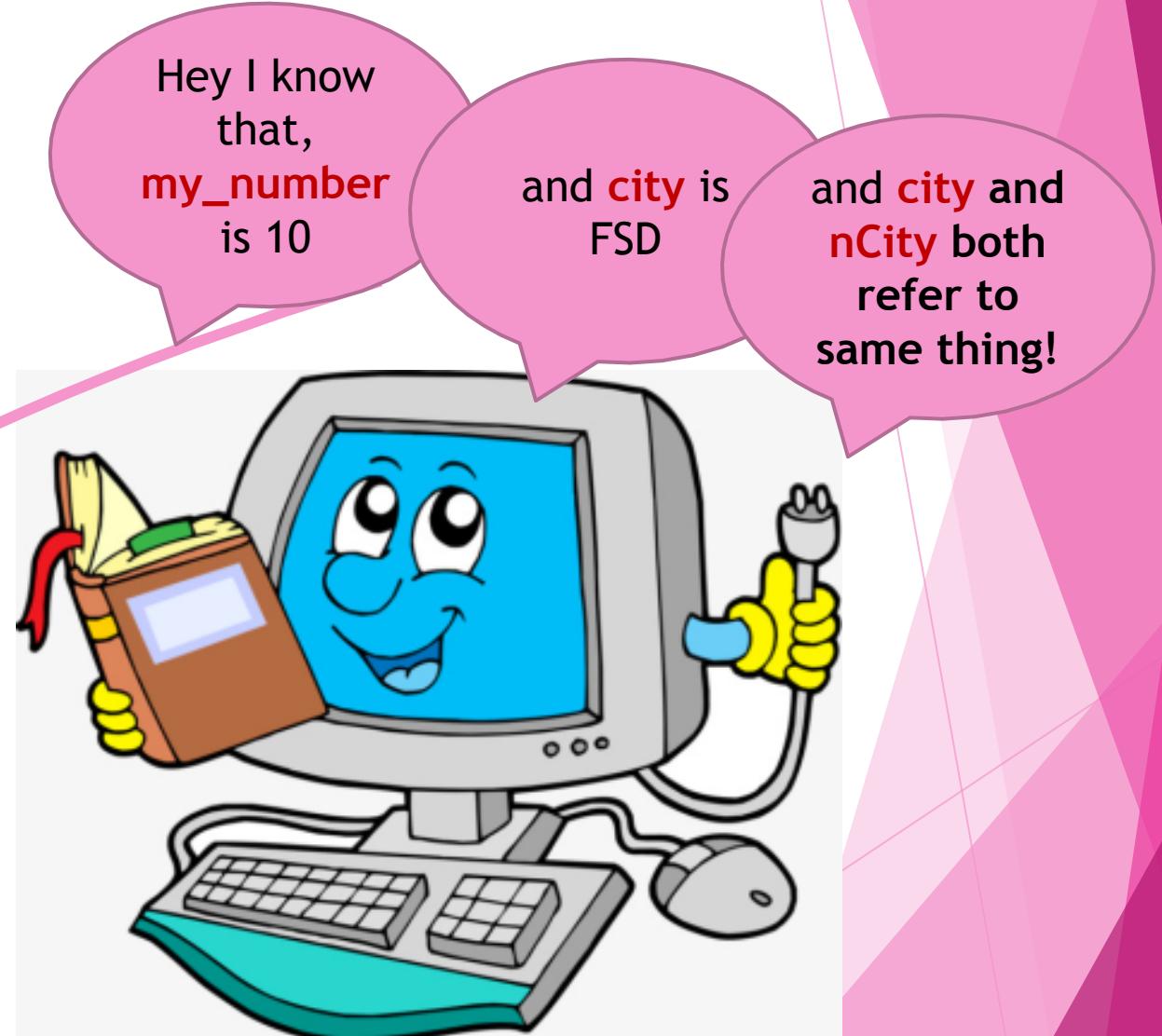
Reading a variable

Setting a variable

```
my_number = 10  
city = 'FSD'  
nCity = city
```

Reading a variable

```
my_number  
city
```



Naming conventions

- ▶ Variable names should be meaningful.
- ▶ Should not start with number, # or &
- ▶ Should not have spaces
- ▶ Camel case wording is better

What is a Camel Case?

`getsystemarea`

The above word is a composition of three words i.e. get, system and area

The 1st word's letter should be small and all further words should start with a capital letter
Example> `getSystemArea`

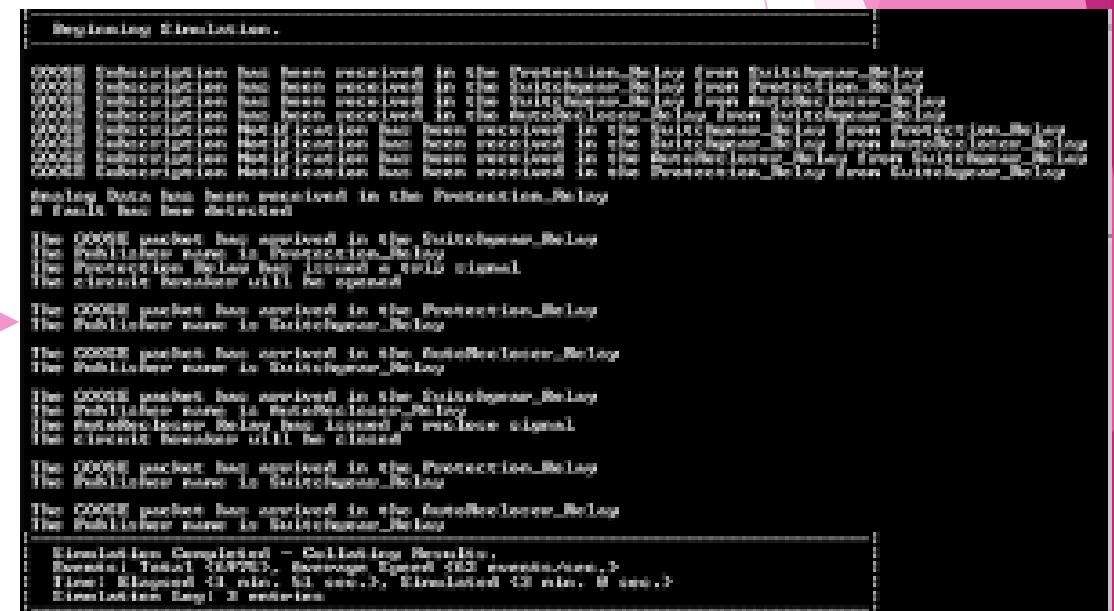


Display data to output

One of the many output options, developers love to use `print` for developing and debugging purposes.

Python program

```
print(my_number)
```



Using print

Displaying multiple values to the output

```
name = "Gamica"
```

```
city = "FSD"
```

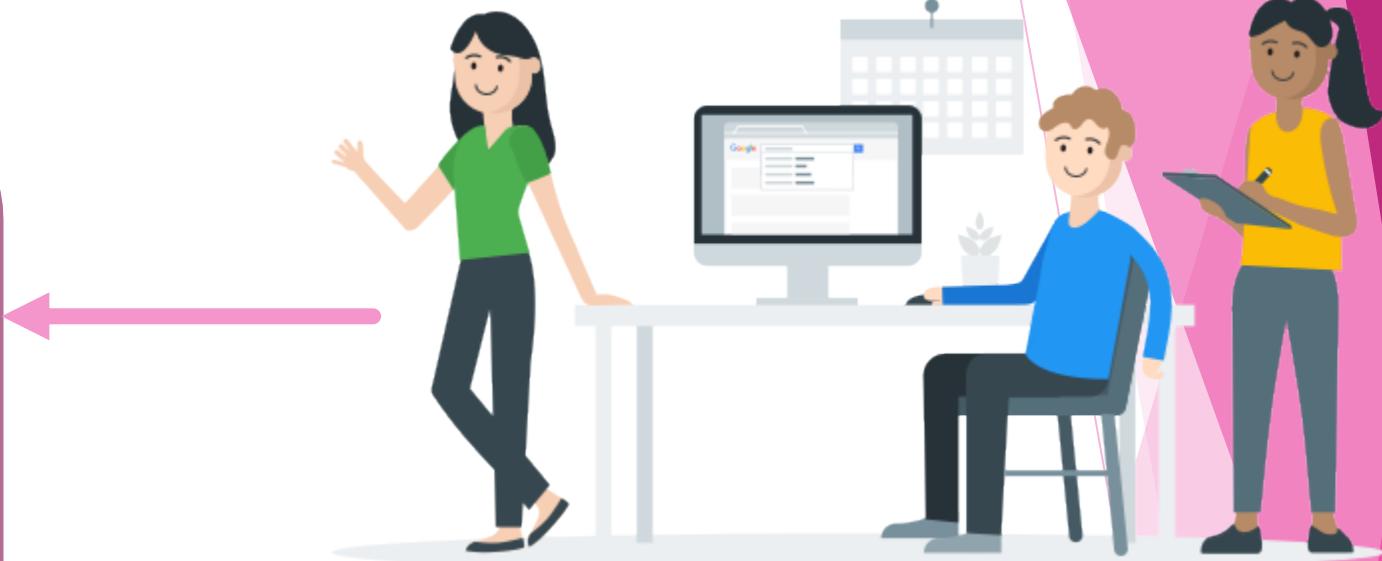
```
print(name, city)
```

Getting input from user

Python program

```
name=input('Enter name')
```

```
//Now display it again to the user  
print(name)
```



```

Beginning Simulation.

0001 Information has been received in the Protection_Relay from Switchgear_Relay
0002 Information has been received in the Protection_Relay from theollector_Relay
0003 Information has been received in the BusbarRelay_Relay from theollector_Relay
0004 Information has been received in the BusbarRelay_Relay from Protection_Relay
0005 Information has been received in the BusbarRelay_Relay from theollector_Relay
0006 Information has been received in the BusbarRelay_Relay from Switchgear_Relay
0007 Information has been received in the BusbarRelay_Relay from theollector_Relay
0008 Information has been received in the Protection_Relay from Switchgear_Relay
0009 Information has been received in the Protection_Relay from theollector_Relay

Scaling Data has been received in the Protection_Relay
#Fault has been detected

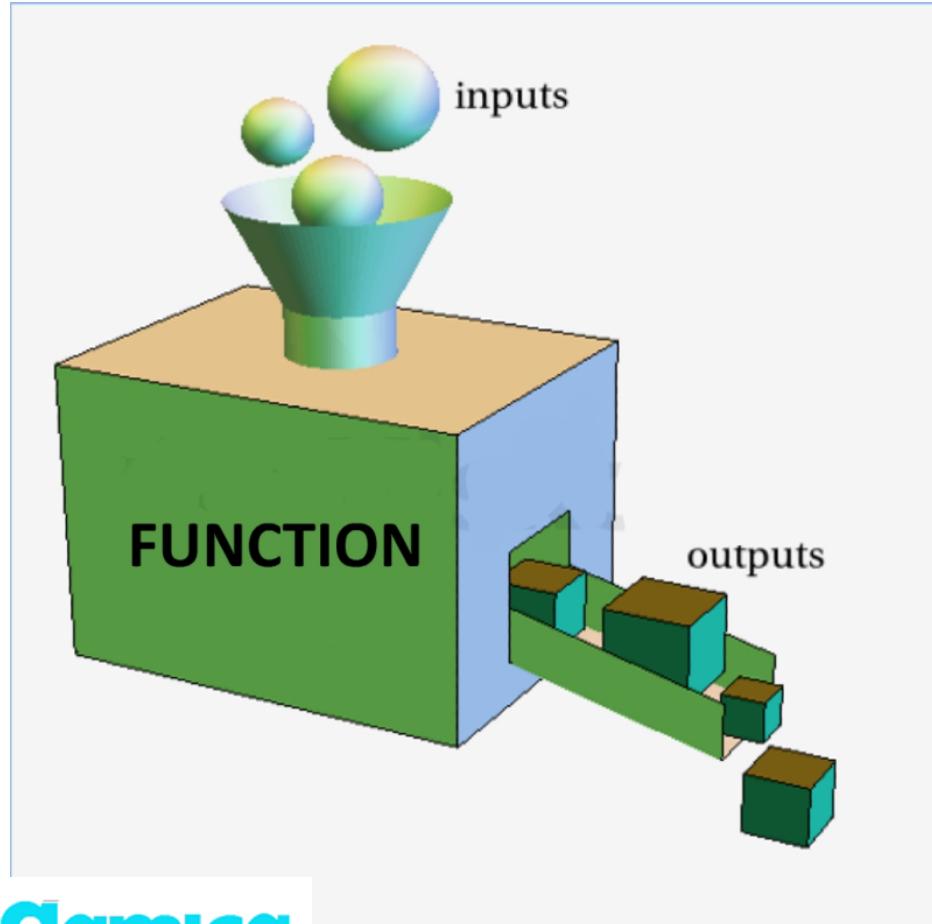
The GOOSE packets have arrived in the Switchgear_Relay
The Publication name is Protection_Relay
The Protection_Relay has issued a trip signal
The circuit breaker will be opened

The GOOSE packets have arrived in the Protection_Relay
The Publication name is Switchgear_Relay
The GOOSE packets have arrived in the BusbarRelay_Relay
The Publication name is Switchgear_Relay
The GOOSE packets have arrived in the BusbarRelay_Relay
The Publication name is Protection_Relay
The Protection_Relay_Relay has issued a reclose signal
The circuit breaker will be closed

The GOOSE packets have arrived in the Protection_Relay
The Publication name is Switchgear_Relay
The GOOSE packets have arrived in the BusbarRelay_Relay
The Publication name is Switchgear_Relay

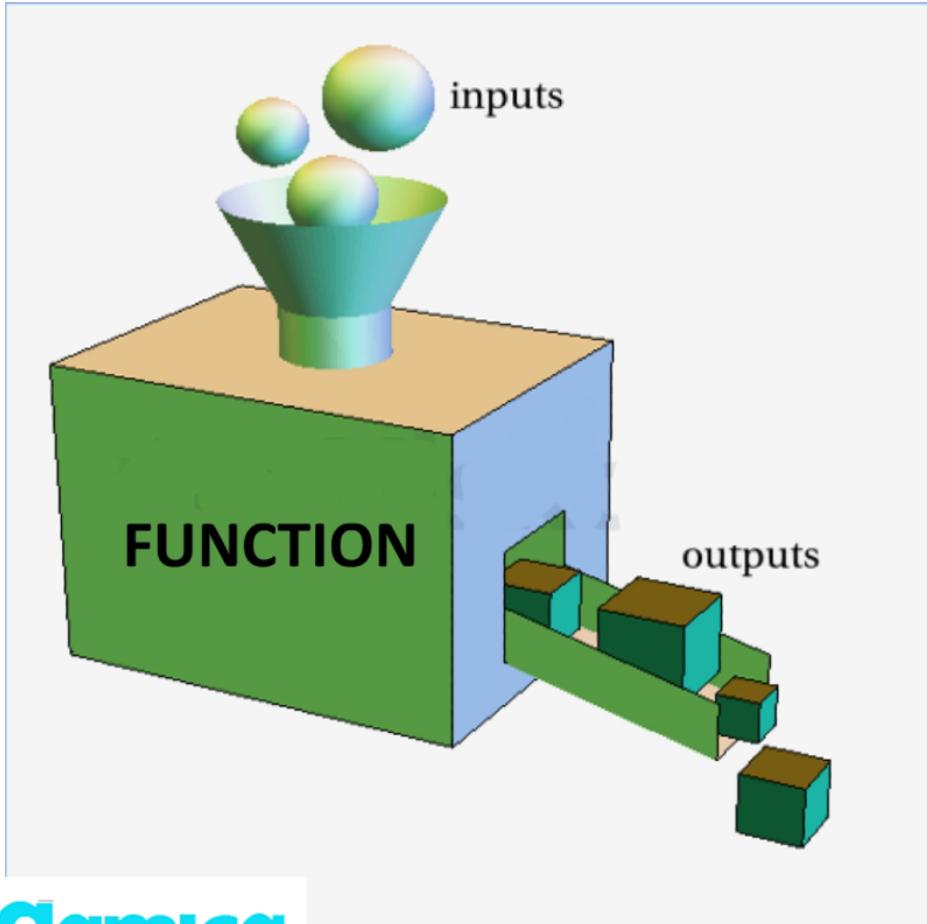
Simulation Completed - Collected Results:
Received Total: 100000, Average Speed: 1000 us/sec.,>
Received Total: 100000, Average Speed: 1000 us/sec.,>
Simulation End, 2 entries
```

Functions



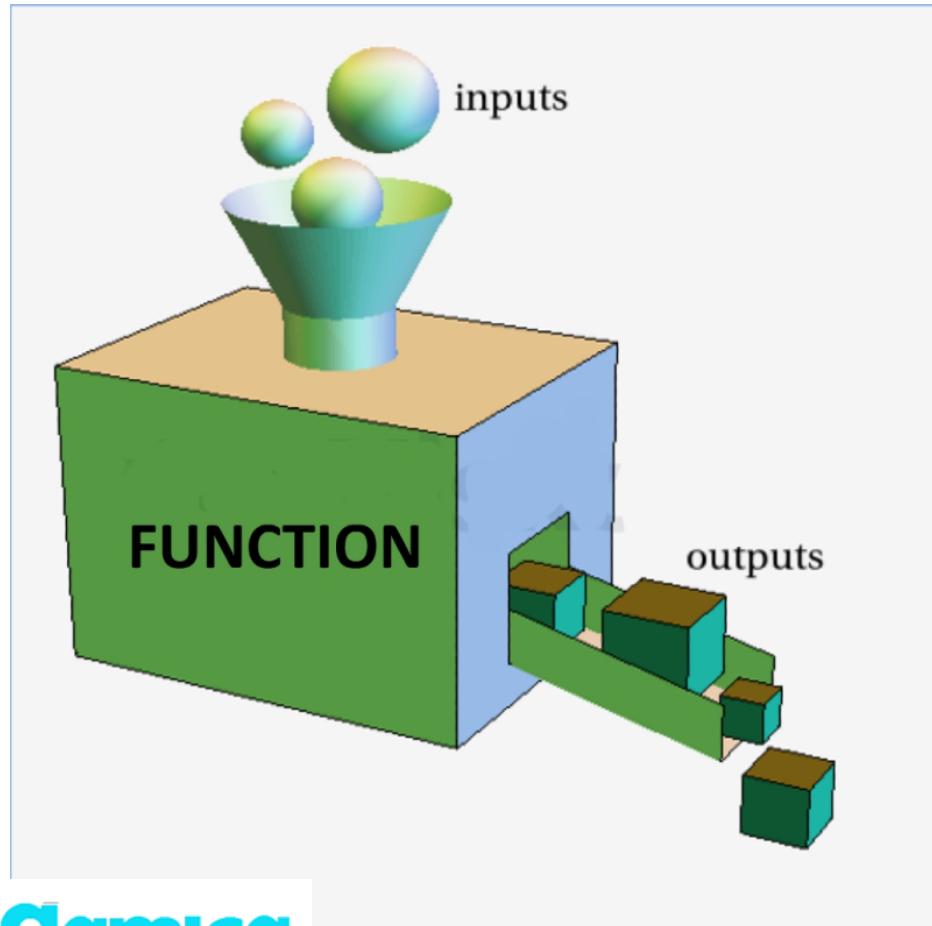
Functions are like **named machines** which have a specific purpose and which can optionally take inputs and may also have outputs.

Functions



Named machine, how?

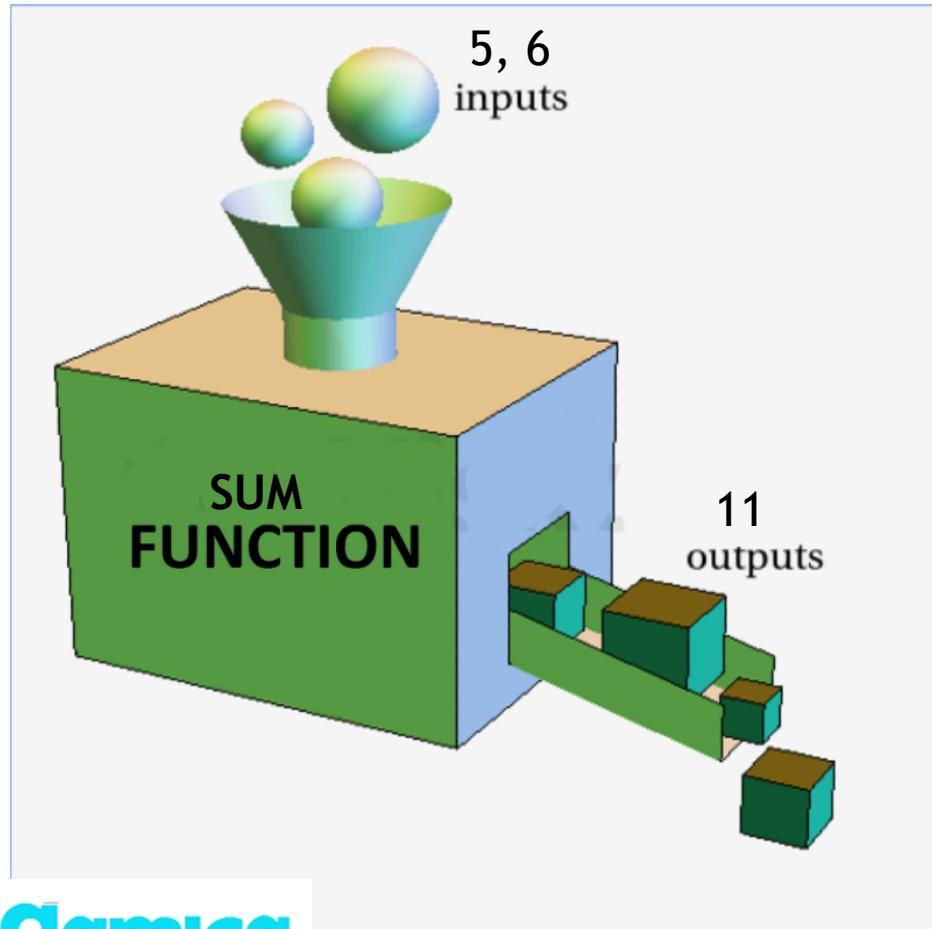
Functions



Suppose we want to add a new feature to our Python code which can take two numbers and output their sum.

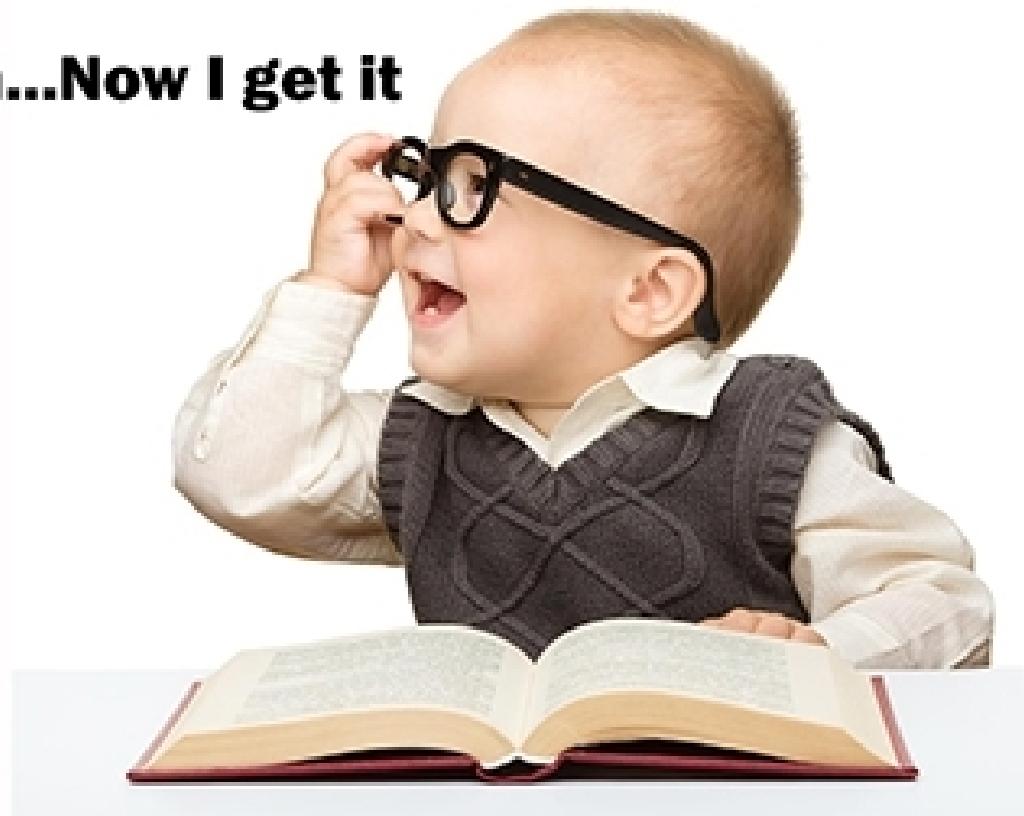
Now consider the addition logic as machine while the numbers its input and their sum as output.

Functions



gamica
cLOUD

Oh...Now I get it



But how to create this machine/function?

define

```
def sum(num1, num2):
```

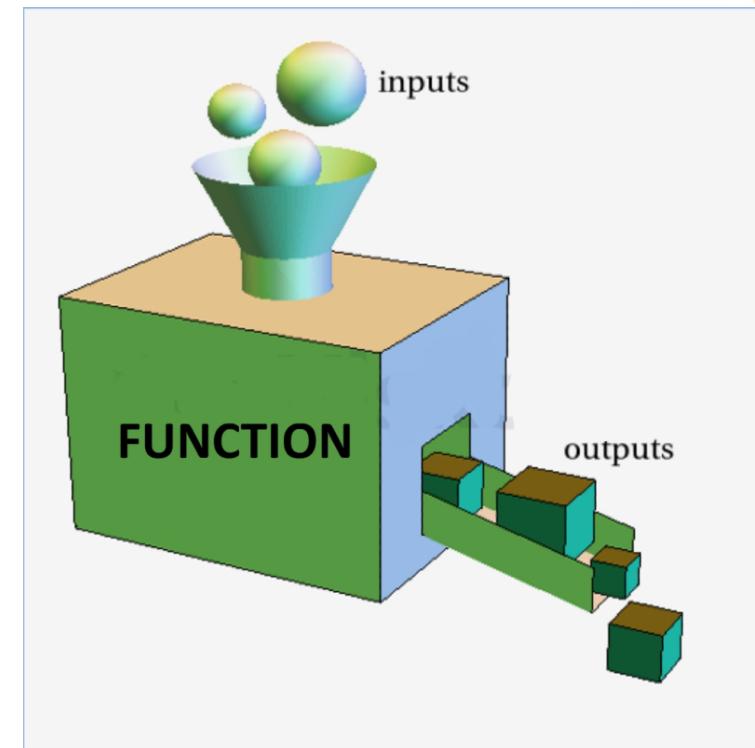
Function name

Two inputs (Leave them empty
if no input required)

```
res = num1 + num2  
print(res)
```

return res

Output



Using(calling) a function

```
my_result = sum(20, 30)
```

```
def sum(num1, num2):
```

```
    res = num1 + num2
```

```
    print(res)
```

```
    return res
```

This 50 will generate back to
the location where sum()
Was called)

Using(calling) a function

```
def sum(num1, num2):  
    return num1 + num2
```

```
def square(num):  
    return num * num
```

```
res = square(5) + sum(10, 20)  
res = 25 + 30
```

```
res = 55
```

Using(calling) a function

```
def sum(num1, num2):  
    return num1 + num2
```

```
def square(num):  
    return num * num
```

```
res = square(5) + sum(10, 20)  
res = 25 + 30
```

```
res = 55
```

Play Time!



Thank You

<https://www.facebook.com/gamicacloud>

